

AVL 树与 Splay 树查找时间对比实验报告

学生姓名：_____ 杨景凯 _____

学 号：_____ 520021910550 _____

2022 年 3 月 22 日

目录

1 实验介绍	3
1.1 实验数据	3
1.2 实验方法	3
2 实验结果	4
3 实验结论	4
3.1 结果分析	4
3.2 理论分析	4

1 实验介绍

1.1 实验数据

实验需要首先对 AVL 树和 Splay 树插入数据，并从插入的数据中寻找一个数据子集作为查找集，对查找集中的数据随机进行一定次数的访问。

设插入的数据量为 n ，要对数据结构查找次数为 m ，其中查找集的数据量为 k 。

在本次实验中，我选取的插入的数据量 n 为 200 (0 至 199)，查找次数 m 为 20000，查找集数据 k 为从 1 开始的分别占总数据 n 比例为 5%，25%，45%，65%，85% 的值。即 10, 50, 90, 130, 170。

1.2 实验方法

首先先将数据 0 至 199 由随机排列的方式插入 AVL 树和 Splay 树。再分别对 AVL 树和 Splay 树做 m 次查找，每次查找为查找从 1 开始的 k 个元素。分别统计 AVL 树和 Splay 树进行所有查找所需的时间。

2 实验结果

实验结果如下表所示：

查找比例	5%	25%	45%	65%	85%
AVLTree	0.018014	0.066067	0.110361	0.164077	0.217001
SplayTree	0.017209	0.112802	0.215909	0.307465	0.394576

3 实验结论

3.1 结果分析

通过实验结果可以发现，在查找数据量很小，查找次数很多的情况下（即 $k \ll n \ll m$ 时），Splay 树的查找时间要短于 AVL 树，但是其他情况下均为 AVL 树的查找时间短于 Splay 树。

随着 k 增加，Splay 树相比于 AVL 树优势逐渐减小最终变为劣势，且劣势逐渐增加，AVL 树优势越来越明显。

因此，在查找数据量很小，查找次数很多的情况下（即 $k \ll n \ll m$ 时），应该首先选择使用 Splay 树，在其余情况下，应该首先选择使用 AVL 树。

3.2 理论分析

上述结果的产生，是由于 AVL 树和 Splay 树的特性决定的。

AVL 是自平衡树，在进行搜索时，对所有数据的搜索时间是接近的。

但是 Splay 树不是自平衡树，它会将近期搜索过的数据放于离根节点较近的位置，缩短再次对该数据搜索的时间。在进行搜索时，不同数据的搜索时间存在较大的差异。如果搜索的数据很久都没有搜索过，那么搜索时间会比较长。但是如果搜索的数据近期被搜索过，那么搜索时间会比较短。

因此，针对于频繁对少量数据的搜索，Splay 树体现了对 AVL 树的劣势。